

Nathalia Bonani

Mestranda em Ecologia, Instituto de Ciências Biológicas, USP.



Currículo Lattes: <http://lattes.cnpq.br/0418436733521485>

Interesses: Ecologia vegetal, germinação de sementes, ecossistemas inflamáveis, manejo e biologia da conservação.

Exercícios

Link para a página com os meus exercícios resolvidos: [Exercícios](#)

Trabalho final

PROPOSTA A

Contextualização:

Pesquisas com germinação de sementes são utilizadas em diversas áreas, principalmente para estudos biológicos, ecológicos e agrônômicos. Os testes controlados em laboratório muitas vezes visam aplicação de distintos tratamentos e acompanhamento do desempenho da(s) espécie(s) alvo(s), geralmente refletido nas taxas finais de germinação e no desempenho desse processo. Seja para testar efeitos da toxicidade de substâncias químicas em organismos vegetais, efeitos de luz, temperatura, nutrientes, tipos de solos ou características intrínsecas das espécies, as planilhas de acompanhamento das respostas germinativas possuem quase sempre a mesma formatação, relatando o sucesso ou fracasso em germinar de cada semente durante um determinado período. Comumente os dados são todos colocados em documentos do tipo excel, bem como os cálculos necessários, mas sua manipulação constante pode ocasionar problemas recorrente em fórmulas e formatação de células. Nesse sentido, a função proposta pretende facilitar a obtenção dos cálculos e avaliações primárias mais utilizadas para experimentos laboratoriais envolvendo sementes, agilizando o processo de análise.

A partir de um data frame padrão contendo dados básicos de acompanhamento de germinação, a função poderá calcular a porcentagem de sementes germinadas (germinability) e viáveis ao final do experimento, os valores médios de germinação para cada tratamento, bem como o tempo médio de germinação (Mean Germination Time) obtido pela seguinte fórmula:

$$\bar{T} = \sum_{i=1}^k n_i t_i / \sum_{i=1}^k n_i$$

Onde t_i : tempo do início do experimento até a observação i (dia); n_i : número de sementes germinadas no tempo i (não o número acumulado, mas o número correspondente à observação), e k : último tempo de germinação.¹

Além disso, poderá gerar um gráfico do tipo boxplot a partir das médias por tratamento e um teste de permutação para verificar se há diferença significativa entre tratamentos.

Planejamento da função:

- **Entrada**

Formato da função: seedR(dados, calc, grafico, teste)

dados= data.frame contendo dados referentes ao A) tipo de tratamento, B) Ob1...Obn= número de sementes germinadas da primeira até n observação, C) Dias em que foram feitas as observações

calc = cálculo que o usuário deseja efetuar. Opções: PG- Porcentagem de germinadas; GT- Tempo médio de germinação.

grafico= TRUE ou FALSE. Se TRUE, a função elabora um gráfico do tipo boxplot com os dados de média de germinação ou de viabilidade nos diferentes tratamentos

teste= TRUE ou FALSE. SE true a função faz um teste de permutação entre os tratamentos para verificar se existe diferença significativa.

- **Pseudo-código**

1. Ler o data.frame
2. Inserir controles de fluxo (verificar dados de entrada)
3. Criar objetos que serão retornados como colunas adicionadas ao data.frame final
4. Indexar colunas para cálculos
5. Fazer o cálculo das porcentagens e médias
6. Indexar o resultado em novas colunas
7. Plotar boxplot com dados das médias por tratamento
8. Salvar o gráfico
9. Fazer teste de reamostragem (for)
10. Fazer cálculo do germination time
11. Indexar o resultado em nova coluna
12. Retornar objetos criados como colunas adicionadas ao data frame
13. Encerrar a função

- **Saída:** *

A função retornará um `data.frame` com a informação inicial e outras colunas contendo o valor calculado das porcentagens, tempo médio de germinação, médias por tratamento, o p-valor do teste de permutação e um gráfico tipo `boxplot` comparando médias

PROPOSTA B

Contextualização:

A procura pela substituição de alimentos de origem animal e a adesão às dietas vegetarianas mais ou menos restritas tem aumentado no Brasil nos últimos anos [2](#). Existe, porém certa dificuldade dos consumidores em escolher adequadamente substituições de alimentos a fim de conseguir resultados equivalentes em receitas tradicionais. Tendo em vista tal dificuldade, essa função foi pensada para pessoas que desejam começar a substituir produtos origem animal (carne, leite, ovos, mel e gelatina) por produtos de origem vegetal, filtrando essa seleção por custo (alto/médio/baixo), tipo de receita (doce/salgada) e restrição alimentar extra (sem glúten/soja), assim o usuário poderá selecionar opções mais adequadas que são frequentemente utilizadas por veganos/vegetarianos.

Planejamento da função

Entrada: `veganizar(animal, custo, restrito)`

`animal` = tipo de ingrediente de origem animal a ser substituído. Nessa opção o usuário poderá escolher entre: carne, frango, peixe, leite, ovo, mel e gelatina.

`custo` = custo do substituto a ser utilizado. As faixas de preço para divisão entre baixo, médio e alto serão definidas pela criadora da função, mas explicitadas para o usuário.

`restrito` = opção para usuário que tenha restrições a produtos de origem vegetal geralmente alergênicos como glúten e soja. Default será `NULL` e opções serão `gluten/soja`.

Verificação dos argumentos (controle de fluxo):

`animal`: Entrada do usuário possui algum dos argumentos possíveis? Se não, retornar mensagem: "Você deve colocar um ingrediente de origem animal válido"

`custo`: Entrada do usuário possui algum dos argumentos possíveis? Se não, retornar mensagem: "Você deve selecionar uma categoria de custo válida"

`restrito`: Entrada do usuário possui algum dos argumentos possíveis? Se não, retornar mensagem: "Você deve selecionar um tipo de restrição válida"

Pseudo-código:

- Criar objetos:
- Substitutos - `data.frame` com colunas contendo todas as opções de substitutos para cada um dos argumentos de `animal` (ou criar vários vetores, um para cada argumento de `animal`?)
- Categorias de custo: vetor composto por caracteres "baixo", "médio" e "alto"
- Restrito: vetor composto por caracteres "glúten", "soja"
- Utilizar `if` e `else` para fazer combinações possíveis entre vetores '`substitutos`'-'`custo`'-'`restrito`'
- Salvar resultado da combinação em um objeto "sugestão"

- Retornar o objeto “sugestão” para visualização do usuário
 - Encerrar função
-

Referências

- 1) RANAL, M. A.; SANTANA, D. G. How and why to measure the germination process? *Revista Brasileira de Botânica*, São Paulo, v.29, n.1, p.1-11. 2006. Acesso em: http://www.scielo.br/scie5cript=sci_arttext&pid=S0100-84042006000100002. Acesso em 17/04/2018.
- 2)DINO. Mudança de hábito dos brasileiros amplia mercado vegetariano e vegano. Disponível em: <https://exame.abril.com.br/negocios/dino/mudanca-de-habito-dos-brasileiros-amplia-mercado-vegetariano-e-vegano/>. Acesso em 30/04/2018

A função da proposta B não parece bem clara. Seu pseudo-código mostra o que vai entrar, mas não o que vai sair, e pelo que está escrito eu não consigo avaliar a dificuldade das operações que vc pretende fazer com o if(), então seria legal vc detalhar seu pseudo-código com essas etapas de cálculos e manipulação das entradas da função.

Achei a proposta A simples, e também não entendi como ela pode ser generalizável para muitos outros usuários.

Meu conselho é que vc detalhe mais o plano B e realize esta mesma proposta. Opcionalmente você pode aumentar quão geral é a proposta A, mas eu não sei se existe tanta margem pra isso nesse assunto.

Para ajudar um pouco mais nesse detalhamento, no dia 15 ou 16 vou entrar aqui de novo pra olhar as modificações do seu pseudo código e te dar mais feedbacks. Depois desse prazo, vc também pode tentar me contatar por whatsapp (por favor não mande áudio) (11) 9-9199-3842

Matheus Januario

Olá, Matheus. Agradeço pelas sugestões.

Tentei explicitar um pouco melhor a generalização do uso de minha proposta A. Entendo que ela está relativamente simples para um usuário avançado, mas ainda não possui muita experiência com o R e então optei aqui propor algo executável ao mesmo tempo que desafiador para mim. Talvez possa acrescentar alguma análise exploratória extra

caso você tenha alguma sugestão. Na minha perspectiva, essa é a proposta mais razoável.

Com relação a proposta B, admito que ainda tenho dificuldade em pensar nos vetores para manipulação que deveriam ser criados para dar conta de todas as combinações entre tipo de substituto-custo-restrição. Mas a ideia geral é criar um “filtro” onde o usuário possa ter no final uma lista de opções alimentares de acordo com as seleções que ele coloca na função. Se você estiver disposto a discutir melhor o pseudo-código comigo, talvez pessoalmente, posso persistir nessa proposta.

Conto com sua compreensão e aguardo seu posicionamento.
Nathalia Bonani

Oi Nathalia me desculpe a demora, mas eu estava esperando que vc me contatasse (eu deixei meu contato no celular). É meio lento nos falarmos por essa wiki. Me mande uma mensagem, por favor.

Realmente a proposta A está bem mais clara pra mim agora mas ainda não sei ainda se ela funciona em usos generalizados, ou se só atende um número muito limitado de pessoas. Lembre que é um critério de correção importante que ela seja generalizável.

Justificativa - Conteúdo inconcluso - justificativa por e-mail

Link para função: [Função](#) Link para o help: [Página de Ajuda](#) Arquivo de exemplo: [teste.csv](#)

From:
<http://ecor.ib.usp.br/> - **ecoR**

Permanent link:
http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:r2018:alunos:trabalho_final:nathalia.bonani:start 

Last update: **2020/09/23 17:15**